

①9 RÉPUBLIQUE FRANÇAISE
INSTITUT NATIONAL
DE LA PROPRIÉTÉ INDUSTRIELLE
PARIS

①1 N° d publication :
(à n'utiliser que pour les
commandes de reproduction)

2 770 662

②1 N° d'enregistrement national : 97 13757

⑤1 Int Cl⁶ : G 06 F 9/38, G 06 F 9/32

⑫

DEMANDE DE BREVET D'INVENTION

A1

②2 Date de dépôt : 03.11.97.

③0 Priorité :

⑦1 Demandeur(s) : INSIDE TECHNOLOGIES SOCIETE
ANONYME — FR.

⑦2 Inventeur(s) : BOUYOUX ERIC.

④3 Date de mise à la disposition du public de la
demande : 07.05.99 Bulletin 99/18.

⑤6 Liste des documents cités dans le rapport de
recherche préliminaire : Se reporter à la fin du
présent fascicule

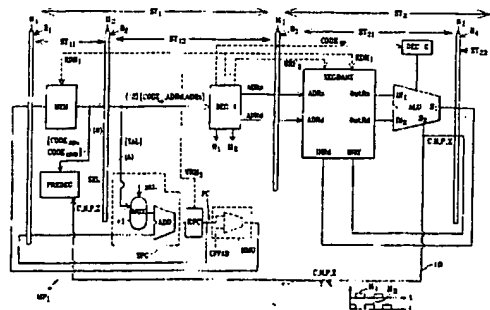
⑥0 Références à d'autres documents nationaux
apparentés :

⑦3 Titulaire(s) :

⑦4 Mandataire(s) : MARCHAND ANDRE.

⑤4 MICROPROCESSEUR PIPELINE A SAUT CONDITIONNEL EN UN CYCLE D'HORLOGE.

⑤7 La présente invention concerne un microprocesseur pipeline (MP1) comprenant, agencés dans des étages pipeline, un compteur ordinal (PC), des moyens (MUX, ADD) d'incrémentation du compteur ordinal (PC) à chaque cycle d'horloge (H1), des moyens de décodage (PREDEC, DEC2) d'un jeu d'instructions du microprocesseur, comprenant des moyens (PREDEC) de décodage d'une instruction de saut conditionnel (JMPC) du compteur ordinal, une banque de registres (REGBANK), une unité de calcul (ALU) comportant une première sortie (S1) pour délivrer un résultat et une deuxième sortie (S2) pour délivrer des bits d'état (C, N, P, Z) du résultat. Selon l'invention, l'unité de calcul (ALU) et les moyens (PREDEC) de décodage de l'instruction de saut conditionnel (JMPC) sont agencés dans deux étages pipeline (ST1, ST2) voisins, et les moyens (PREDEC) de décodage de l'instruction de saut conditionnel (JMPC) sont connectés à la deuxième sortie (S2) de l'unité de calcul (ALU) - Avantages: exécution d'une instruction de saut conditionnel (JMPC) en un cycle d'horloge.



FR 2 770 662 - A1



MICROPROCESSEUR PIPELINE A SAUT CONDITIONNEL EN UN CYCLE
D'HORLOGE

La présente invention concerne un microprocesseur du type pipeline comprenant un compteur ordinal, un circuit d'incrémentation du compteur ordinal, des moyens de décodage d'un jeu d'instructions du microprocesseur
5 comprenant des moyens de décodage d'une instruction de saut conditionnel du compteur ordinal, et une unité de calcul comportant une première sortie pour délivrer un résultat et une deuxième sortie pour délivrer des bits d'état du résultat.

10 La présente invention concerne également l'exécution d'une instruction de saut conditionnel dans un microprocesseur pipeline du type ci-dessus.

Par "instruction de saut conditionnel", on désigne dans la présente demande une instruction JMPc de saut du
15 compteur ordinal PC ("program counter") d'un microprocesseur, dont l'exécution dépend du résultat d'une instruction précédente. Par exemple, les instructions suivantes sont des sauts conditionnels :

- 1) JMPc IF C=0 VAL (saut de valeur VAL si C=0),
- 20 2) JMPc IF N=0 VAL (saut de valeur VAL si N=0),
- 3) JMPc IF P=0 VAL (saut de valeur VAL si P=0),
- 4) JMPc IF Z=0 VAL (saut de valeur VAL si Z=0),

le bit C ("carry") étant le bit de report de somme (ou de débordement), le bit N le bit de signe, le bit P le bit
25 de parité et le bit Z le bit de valeur nulle du résultat d'une opération réalisée par une unité de calcul arithmétique et logique ALU. Ces bits C, N, P, Z, ou drapeaux ("flags"), sont stockés classiquement dans un registre désigné Rf ("flags register").

30 Dans l'art antérieur, une instruction de saut conditionnel ne peut être exécutée avant que le résultat

de l'instruction précédente soit connu et que le registre Rf soit mis à jour.

Par exemple, la séquence de programme suivante :

(1) (ADR₁) SUB R1,R2,R3

5 (2) (ADR₂) JMPc IF Z=1 VAL

qui signifie :

(1) "soustraire le contenu d'un registre R3 au contenu d'un registre R2, et enregistrer le résultat dans un registre R3",

10 (2) "saut du programme à l'adresse ADR₂ + VAL si résultat nul, sinon aller à l'adresse immédiatement suivante ADR₃ du programme",

implique les étapes de traitement suivantes, chaque étape étant réalisée en un coup d'horloge :

15 1.1 - lecture de l'instruction SUB à l'adresse ADR₁ de la mémoire programme du microprocesseur, et incrémentation du compteur ordinal PC (ADR₁ devient ADR₂),

1.2 - décodage de l'instruction SUB,

1.3 - lecture des registres R2 et R3,

20 1.4 - soustraction par l'unité ALU du contenu du registre R3 au contenu du registre R2,

1.5 - enregistrement du résultat de la soustraction dans le registre R1 et enregistrement dans le registre Rf des drapeaux C,N,P,Z délivrés par l'unité ALU,

25 2.1 - lecture de l'instruction JMPc à l'adresse ADR₂ et incrémentation normale du compteur ordinal (ADR₂ devient ADR₃),

2.2 - décodage de l'instruction JMPc,

2.3 - lecture du drapeau Z dans le registre Rf ; si Z=1,

30 calcul de l'adresse ADR₂ + VAL et chargement de cette adresse dans le compteur ordinal PC.

En principe, et conformément au fonctionnement pipeline, les étapes de traitement de deux instructions successives se chevauchent avec un décalage d'un rang.

35 Dans la séquence ci-dessus, on voit toutefois que l'étape

2.3 ne peut chevaucher l'étape 1.4 car le résultat de l'opération précédente n'est pas encore disponible dans le registre Rf. Egalement, une nouvelle instruction suivant l'instruction JMPc ne peut être lue avant que
5 l'étape 2.3 de l'instruction JMPc ne soit terminée car l'on ne sait pas encore à quelle adresse cette instruction doit être lue.

Ainsi, la solution classique pour pallier cet inconvénient consiste à suspendre l'injection d'une
10 nouvelle instruction dans la chaîne pipeline jusqu'à ce que les drapeaux C, N, P, Z soient chargés dans le registre Rf (étape 1.5) et que l'instruction de saut conditionnel soit traitée (étape 2.3).

Cette méthode présente l'inconvénient de bloquer
15 temporairement le fonctionnement de la chaîne pipeline et de ralentir l'exécution d'un programme. De façon générale, le temps perdu pour l'exécution d'un programme est proportionnel au nombre de sauts conditionnels que contient le programme.

20 La présente invention a pour principal objectif de prévoir une structure de microprocesseur pipeline permettant d'exécuter une instruction de saut conditionnel sans interruption de la chaîne de traitement pipeline.

25 Un objectif secondaire de la présente invention est de prévoir une structure de microprocesseur pipeline qui soit simple et consomme peu de courant.

Pour atteindre ces objectifs, la présente invention prévoit un microprocesseur du type cité ci-dessus dans
30 lequel l'unité de calcul et les moyens de décodage de l'instruction de saut conditionnel sont agencés dans deux étages pipeline voisins, et les moyens de décodage de l'instruction de saut conditionnel sont connectés à la deuxième sortie de l'unité de calcul.

Selon un mode de réalisation, le microprocesseur comprend un premier et un deuxième étages pipeline comprenant chacun un premier et un deuxième secteurs, le premier secteur de chaque étage étant actif pendant une
5 première demi-période d'horloge et le deuxième secteur de chaque étage actif pendant une deuxième demi-période d'horloge.

Selon un mode de réalisation, les moyens de décodage de l'instruction de saut conditionnel sont
10 agencés dans le premier secteur du premier étage pipeline, l'unité de calcul est agencée dans le premier secteur du deuxième étage pipeline.

Selon un mode de réalisation, les moyens de décodage de l'instruction de saut conditionnel sont
15 agencés dans le deuxième secteur du premier étage pipeline, l'unité de calcul est agencée dans le premier secteur du deuxième étage pipeline.

Selon un mode de réalisation, le circuit d'incrémentement du compteur ordinal est agencé dans le
20 deuxième secteur du premier étage pipeline.

Avantageusement, le microprocesseur comprend une banque de registres appartenant en lecture au premier secteur et en écriture au deuxième secteur du deuxième étage pipeline. Cette disposition permet de simplifier la
25 structure du microprocesseur qui peut ne comprendre que deux étages pipeline, au bénéfice de la consommation électrique, et exécuter des instructions compactes comprenant uniquement un code opération, l'adresse d'un registre source et l'adresse d'un registre de destination
30 du résultat de l'opération désignée par l'instruction.

Selon un mode de réalisation, les secteurs du deuxième étage pipeline sont délimités par un tampon à trois états agencé entre les première et deuxième sorties de l'unité de calcul et des entrées d'écriture de la
35 banque de registre.

Selon un mode de réalisation, l'instruction de saut conditionnel est une instruction de saut relatif, les moyens d'incrémentation du compteur ordinal comprennent un circuit additionneur recevant en entrée la valeur
5 actuelle du compteur ordinal et la sortie d'un circuit multiplexeur, le circuit multiplexeur est piloté par les moyens de décodage de l'instruction de saut conditionnel et reçoit en entrée une valeur d'incrémentation normale du compteur ordinal et une valeur de saut relatif du
10 compteur ordinal, contenue dans l'instruction de saut conditionnel.

Selon un mode de réalisation, l'instruction de saut conditionnel est une instruction de saut absolu, les moyens d'incrémentation du compteur ordinal comprennent
15 un circuit multiplexeur recevant en entrée la sortie d'un circuit additionneur et une valeur de saut absolu du compteur ordinal contenue dans l'instruction de saut conditionnel, le circuit multiplexeur est piloté par les moyens de décodage de l'instruction de saut conditionnel,
20 et le circuit additionneur reçoit en entrée la valeur actuelle du compteur ordinal et une valeur d'incrémentation normale du compteur ordinal.

Selon un mode de réalisation, les moyens de décodage sont agencés pour décoder une instruction
25 compacte de saut conditionnel comprenant un code opération, une condition et une valeur de saut.

Ces caractéristiques et avantages de la présente invention seront exposés plus en détail dans la description suivante de deux exemples de réalisation d'un
30 microprocesseur selon l'invention, faite à titre non limitatif en relation avec les figures jointes parmi lesquelles :

- la figure 1 est le schéma électrique d'un premier mode de réalisation d'un microprocesseur selon l'invention, et

- la figure 2 représente une variante de réalisation du microprocesseur de la figure 1.

La figure 1 représente un microprocesseur pipeline MP1 selon l'invention. Le microprocesseur est organisé pour exécuter des instructions compactes se présentant sous le format suivant :

CODE _{OP}	ADRd	ADRs	-
--------------------	------	------	---

Ces instructions, codées ici sur 16 bits, comprennent un champ CODE_{OP} codé sur 4 bits, donnant l'opération OP à exécuter par l'unité ALU, un champ ADRd de 4 bits donnant l'adresse d'un registre Rd de destination du résultat, et un champ ADRs de 4 bits donnant l'adresse d'un registre source Rs. Le dernier champ, de 4 bits, n'est pas utilisé ici. De façon classique, l'opération OP est par exemple l'addition ADD (additionner Rs à Rd et stocker le résultat dans Rd), la soustraction SUB (soustraire Rs à Rd et stocker le résultat dans Rd), le OU logique (stocker dans R1 le résultat de R1 OU R2 calculé bit à bit), le ET logique, etc..

Le microprocesseur est également organisé pour exécuter une instruction compacte de saut conditionnel JMPc, qui se présente par exemple sous le format suivant:

CODE _{JMPc}	CODE _{COND}	VAL
----------------------	----------------------	-----

Cette instruction comprend le code opération CODE_{JMPc} de l'instruction, sur 4 bits, le code CODE_{COND} de la condition du saut COND, sur 4 bits, et la valeur de saut VAL, sur 8 bits. Comme expliqué au préambule, la condition de saut COND est en rapport avec la valeur d'un bit d'état C, N, P, ou Z, et peut par exemple être : IF Z=0, IF Z=1, IF C=0, IF C=1, etc..

Par ailleurs, le microprocesseur comprend les éléments suivants : une banque de registre REGBANK ; une unité de calcul arithmétique et logique ALU ; un compteur ordinal PC pour donner l'adresse de lecture, dans une mémoire programme MEM, des instructions à traiter ; un circuit SPC d'incrémentation, à chaque nouvelle instruction reçue, du compteur ordinal PC ; des moyens DEC1, DEC2, PREDEC de décodage d'un jeu d'instructions du microprocesseur. Les moyens de décodage DEC1, DEC2, PREDEC, l'unité ALU et la banque REGBANK, représentés sous forme de blocs, sont à la portée de l'homme de l'art et leur structure interne ne sera pas décrite. L'unité ALU comporte ici deux entrées IN1, IN2, une sortie de résultat S1 et une sortie S2 délivrant les bits d'état C, N, P, Z du résultat. La banque de registres REGBANK contient divers registres de travail, par exemple seize registres R1 à R16 (non représentés), et des registres hardware parmi lesquels se trouve un registre Rf contenant les bits d'état C, N, P, Z. La banque REGBANK comporte une entrée ADRs pour recevoir l'adresse du registre source Rs, une entrée ADRd pour recevoir l'adresse du registre de destination Rd, une sortie OUTRs de lecture du registre source Rs, une sortie OUTRd de lecture du registre de destination Rd, une entrée INRd pour l'écriture d'un résultat dans le registre de destination Rd, et une entrée INRf pour écrire dans le registre Rf les bits d'état C, N, P, Z d'un résultat. Les moyens de décodage DEC1, DEC2, PREDEC comprennent un décodeur principal DEC1, un décodeur DEC2 pour piloter l'unité ALU, et un décodeur PREDEC de l'instruction de saut conditionnel JMPc.

Le chemin de données du microprocesseur est divisé en deux étages pipeline ST1, ST2 contigus délimités par des registres B1, B3. Ces registres B1, B3, par exemple des registres à verrouillage (ou "latch"), sont

déclenchés par le passage à 1 d'un signal d'horloge H1. Par ailleurs, les deux étages ST1, ST2 sont divisés chacun en deux secteurs ST11/ST12, ST21/ST22 au moyen d'un registre B2 agencé entre les registres B1, B3 et d'un
5 registre B4 disposé après le registre B3 selon le sens de circulation des données. Le registre B2 est par exemple un registre à verrouillage déclenché par le passage à 1 d'un signal H2 déphasé de 180° relativement au signal H1. Le registre B4 est de préférence un registre tampon trois
10 états ("buffer tristate") transparent quand le signal H2 est à 1 et dans l'état haute impédance quand H2 est à 0.

Ainsi, le signal d'horloge H1 synchronise le transfert des données de l'étage ST1 vers l'étage ST2, et que le signal H2 synchronise le transfert des données du
15 premier secteur ST11, ST21 vers le deuxième secteur ST12, ST22 de chaque étage ST1, ST2. Pour fixer les idées, les signaux d'horloge H1 et de transfert H2 sont représentés au bas de la figure 1.

Selon l'invention, l'unité ALU est agencée dans le
20 secteur ST21 de l'étage pipeline ST2 et le décodeur PREDEC agencé dans le secteur ST11 de l'étage ST1. La sortie S2 de l'unité ALU, qui délivre les bits C, N, P, Z, est appliquée par l'intermédiaire d'un bus 10 au décodeur PREDEC. Comme on le verra par la suite, cette
25 connexion du prédécodeur PREDEC à la sortie de l'unité ALU délivrant les bits C, N, P, Z, permet de traiter immédiatement l'instruction de saut conditionnel, sans attendre que ces bits soient chargés dans le registre Rf. Le décodeur PREDEC reçoit les 8 premiers bits des
30 instructions présentes à la sortie de la mémoire MEM. Dans le cas de l'instruction JMPc, ces huit premiers bits comprennent le code opération CODE_{JMPc} et le code condition CODE_{COND}.

Par ailleurs, les sorties OUTRs et OUTRd de la
35 banque REGBANK sont appliquées sur les entrées IN1 et IN2

de l'unité ALU, respectivement. Les sorties S1 et S2 de l'unité ALU sont renvoyées par l'intermédiaire du registre B4 sur les entrées INRd et INRf de la banque REGBANK, respectivement. Le décodeur DEC1, disposé dans le secteur ST12, reçoit au moins les 12 premiers bits des instructions, c'est-à-dire les champs CODE_{Op}, ADRd et ADRs, par l'intermédiaire du registre B2. Les adresses ADRs, ADRd sont transférées sur les entrées correspondantes de la banque REGBANK par l'intermédiaire du registre B3.

La banque REGBANK est commandée en lecture par un signal RDH1 et en écriture par un signal WRH2, ces signaux étant émis par le décodeur DEC1. Le signal de lecture RDH1 est émis quand le signal d'horloge H1 est à 1 et le signal d'écriture WRH2 est émis quand le signal H2 est à 1. La banque REGBANK appartient ainsi, en lecture, au premier secteur ST21 de l'étage ST2 et en écriture au deuxième secteur ST22 de l'étage ST2. Par ailleurs, le décodeur DEC2, agencé dans le secteur ST21, reçoit du décodeur DEC1 les codes CODE_{Op} des opérations que doit réaliser l'unité ALU.

Ici, le compteur ordinal PC est réalisé au moyen d'un registre tampon RPC transparent sur réception du signal d'écriture WRH2. A chaque cycle d'horloge, le tampon RPC reçoit du circuit SPC une nouvelle valeur d'adresse, ou adresse PC. Après application du signal WRH2, l'adresse PC se retrouve à la sortie du tampon RPC et est renvoyée sur l'entrée d'adresse de la mémoire MEM par l'intermédiaire du registre B1. La mémoire MEM délivre en sortie l'instruction contenue à l'adresse PC, sur réception du signal de lecture RDH1 (dans un souci de cohérence du schéma de la figure 1, la mémoire MEM est représentée dans le secteur ST11, bien qu'étant un élément périphérique du microprocesseur). De façon optionnelle, l'adresse PC peut être appliquée à la

mémoire MEM per l'intermédiaire d'un convertisseur MMU ("Memory Management Unit") représenté en traits pointillés sur la figure. Le convertisseur MMU reçoit sur une entrée une adresse offset OFFAD et convertit
 5 l'adresse PC par addition (comme représenté sur la figure) ou par concaténation de l'adresse OFFAD avec l'adresse PC.

Enfin, le circuit SPC comprend un multiplexeur MUX recevant sur une première entrée une valeur
 10 d'incrément normal du compteur ordinal, ici la valeur +1, et sur une deuxième entrée les huit derniers bits de l'instruction présente à la sortie de la mémoire MEM, par l'intermédiaire du registre B2. Selon le format des instructions décrit ci-dessus, ces huit derniers bits
 15 correspondent à la zone VAL de l'instruction JMPc. La sortie du multiplexeur MUX est appliquée sur une entrée d'un additionneur ADD dont la sortie est appliquée en entrée du tampon SPC. L'adresse PC à la sortie du tampon SPC est renvoyée par l'intermédiaire du registre B1 sur
 20 la deuxième entrée de l'additionneur ADD.

Pour illustrer le fonctionnement du microprocesseur de l'invention, on se référera par exemple à la séquence de programme suivante :

- 25 (1) (ADR₁) SUB R1,R2
 (2) (ADR₂) JMPc IF Z=1 VAL

Ces instructions sont délivrées par la mémoire MEM sous la forme suivante :

30

CODE _{SUB}	ADR ₁	ADR ₂	
CODE _{JMPc}	CODE _{IF Z=1}	VAL	

La séquence est exécutée par le microprocesseur selon des étapes 1.1 à 2.2 décrites ci-après. Les diverses opérations réalisées au cours de chacune de ces étapes se déroulent de façon asynchrone en fonction du
5 flot des données à travers les portes logiques des éléments du microprocesseur, et sont considérées comme achevées à la fin de chaque étape.

1.1 - H1 = 1, H2 = 0, registre B1 déclenché, secteur ST11 actif :

- 10 - l'adresse PC (ici ADR_1) est appliquée à l'entrée de l'additionneur ADD et de la mémoire MEM,
- la mémoire MEM reçoit le signal de lecture RDH1 et délivre l'instruction SUB R1, R2,
- le décodeur PREDEC reçoit le code $CODE_{SUB}$ et met le
15 signal SEL à 0 car le code $CODE_{JMPC}$ n'est pas reconnu.

1.2 - H1 = 0, H2 = 1, registre B2 déclenché, secteur ST12 actif :

- le décodeur DEC1 reçoit l'instruction, envoie les adresses ADR_1 et ADR_2 en direction de la banque REGBANK
20 et le code opération $CODE_{SUB}$ en direction du décodeur DEC2 (ces données sont interceptées et bloquées par le registre B3),
- le multiplexeur MUX reçoit le signal SEL (à 0), sélectionne l'entrée "+1", de sorte que la sortie de
25 l'additionneur ADD délivre la valeur ADD_1+1 , soit ADD_2 ,
- le registre RPC reçoit le signal d'écriture WRH2 et l'adresse PC (anciennement ADD_1) devient ADD_2 .

1.3 - H1 = 1, H2 = 0, registre B3 déclenché, secteur ST21 actif :

- 30 - la banque REGBANK reçoit les adresses ADR_1 , ADR_2 et le signal de lecture RDH1, et délivre le contenu des registres R1, R2 à l'unité ALU,
- l'unité ALU effectue la soustraction, délivre le résultat (R1-R2) sur la sortie S1, et les bits C, N, P, Z

sur la sortie S2. Les bits C, N, P, Z sont appliqués au décodeur PREDEC via le bus 10.

1.4 - H1 = 0, H2 = 1, registre B4 transparent, secteur ST22 actif :

- 5 - les sorties S1, S2 de l'unité ALU sont connectées aux entrées INRd, INRf de la banque REGBANK. La banque reçoit le signal d'écriture WRH2. Le résultat de l'opération (R1-R2) est enregistré dans R1 et les bits C,N,P,Z présents à la sortie S2 de l'unité ALU sont enregistrés
- 10 dans Rf.

2.1 - H1 = 1, H2 = 0, registre B1 déclenché, secteur ST11 actif :

- l'adresse PC (maintenant ADR_2) est appliquée en entrée de l'additionneur ADD et de la mémoire MEM,
- 15 - la mémoire MEM reçoit le signal de lecture RDH1 et délivre l'instruction JMPc IF Z=1 VAL,
- le décodeur PREDEC reçoit le code $CODE_{JMPc}$, le code condition $CODE_{IF Z=1}$ et, via le bus 10, les bits C, N, P, Z délivrés par la sortie S2 de l'unité ALU (une période
- 20 flottante intervient, l'unité ALU effectuant au même instant le calcul de l'opération SUB de l'étape 1.3 ci-dessus),
- le décodeur PREDEC met le signal SEL à 1 si la condition Z = 1 est réalisée ou le met à 0 si Z = 0.

25 2.2 - H1 = 0, H2 = 1, registre B2 déclenché, secteur ST12 actif :

- le champ VAL (valeur du saut d'adresse) est transféré à l'entrée du multiplexeur MUX,
- le multiplexeur MUX reçoit le signal SEL et sélectionne
- 30 l'une de ses deux entrées. Si SEL = 1, la sortie de l'additionneur ADD délivre la valeur $ADD_n = ADD_2 + VAL$; si SEL = 0 la sortie de l'additionneur ADD délivre la valeur $ADD_3 = ADD_2 + 1$,

- le registre RPC reçoit le signal WRH2 et l'adresse PC devient égale à ADD_n ou à ADD_3 selon la valeur du signal SEL.

Dans le microprocesseur pipeline selon l'invention, les étapes décrites ci-dessus se chevauchent comme résumé dans le tableau ci-après.

On voit dans ce tableau que la première étape 2.1 de l'instruction JMPc commence au même moment que l'étape 1.3 de l'instruction SUB, c'est-à-dire au deuxième cycle pipeline de cette instruction. On voit également que la première étape 3.1 de lecture d'une nouvelle instruction suivant l'instruction JMPc peut commencer après l'étape 2.2, l'instruction JMPc ayant été exécutée et l'adresse PC mise à jour en un seul cycle pipeline. A la fin de l'étape 2.2, l'adresse PC est en effet égale à ADD_n ou à ADD_3 selon que la condition du saut a été réalisée ou non.

1 cycle pipeline		1 cycle pipeline		1 cycle pipeline	
1/2 cycle	1/2 cycle	1/2 cycle	1/2 cycle	1/2 cycle	1/2 cycle
H1=1	H2=1	H1=1	H2=1	H1=1	H1=1
étape 1.1	étape 1.2	étape 1.3	étape 1.4		
		étape 2.1	étape 2.2		
				étape 3.1	étape 3.2

En définitive, et très avantageusement, aucune interruption de la chaîne pipeline n'est nécessaire pour le traitement de l'instruction JMPc. De plus, la structure du microprocesseur selon l'invention, particulièrement simple, garantit une faible consommation de courant. Une mesure supplémentaire permettant de limiter la consommation électrique consiste à disposer sur les sorties OUTRs, OUTRd de la banque REGBank des registres tampons pilotés par un signal H1' décalé d'une durée Δt relativement au signal H1. De cette manière, l'unité ALU n'est connectée aux sorties OUTRs, OUTRd qu'au moment où celles-ci sont stabilisées. Le signal

décalé H1' peut être obtenu simplement au moyen d'un circuit à retard, par exemple des portes inverseuses en cascade.

La figure 2 représente une variante MP2 du microprocesseur selon l'invention qui se distingue essentiellement du mode de réalisation précédent par le fait que le décodage de l'instruction JMPc est confié au décodeur DEC2, maintenant désigné DEC3. Le décodeur DEC3 réalise ainsi les opérations des décodeurs PREDEC et DEC2 de la figure 1 et délivre notamment le signal SEL au cours de l'étape 2.2, en référence à la séquence décrite ci-dessus. Il convient de noter que le signal SEL peut être délivré rapidement par le décodeur DEC3 au cours de l'étape 2.2 étant donné que les bits C, N, P, Z sont validés à la fin de l'étape 2.1 (qui correspond à l'étape de calcul 1.3 de l'instruction précédente).

Cette variante de réalisation montre par ailleurs que les bits C, N, P, Z délivrés par l'unité ALU peuvent être appliqués de façon synchrone au décodeur DEC3 par l'intermédiaire du registre B4, par exemple au moyen d'un bus 11 représenté en traits pointillés. Ainsi, lorsque l'on dit, dans la présente demande, que l'on connecte la sortie S2 de l'unité ALU aux moyens de décodage de l'instruction JMPc, cela ne signifie pas obligatoirement que la sortie de l'unité ALU est connectée directement aux moyens de décodage. Cela signifie simplement que les bits C, N, P, Z sont prélevés à la sortie de l'unité ALU ou sur le bus qui les transporte vers le registre Rf au lieu d'être lus dans le registre Rf comme dans l'art antérieur.

Bien entendu, la présente invention est susceptible de nombreuses autres variantes de réalisation et perfectionnements. Dans ce qui précède, on a décrit un microprocesseur simplifié ne comportant que les éléments nécessaires à la description et la compréhension de

l'invention. En pratique, le microprocesseur selon l'invention pourra comprendre divers autres moyens, par exemple des moyens de décodage d'instructions complexes, des moyens d'écriture de données dans la mémoire, etc.

5 Notamment, on a décrit dans ce qui précède un microprocesseur exécutant une instruction de saut "relatif" en additionnant une valeur VAL à l'adresse PC. Une variante consiste à réaliser une opération de saut "absolu" en inversant la disposition du multiplexeur MUX
10 et de l'additionneur ADD (non représenté). Dans ce cas, l'additionneur ADD reçoit en entrée l'adresse PC et la valeur d'incrément +1. La sortie de l'additionneur ADD est appliquée sur une entrée du multiplexeur MUX qui reçoit toujours sur son autre entrée la valeur de saut
15 VAL. La sortie du multiplexeur MUX est appliquée au registre RPC. Ainsi, selon cette variante, l'adresse PC est incrémentée de la valeur + 1 ou est portée à la valeur VAL selon que le signal SEL est à 1 ou à 0. La valeur VAL représente, dans ce cas, une adresse de saut
20 absolu et non une valeur de saut relatif.

 Enfin, ces deux modes de réalisation du saut conditionnel absolu ou relatif peuvent être juxtaposés. Dans ce cas, l'agencement représenté en figures 1 et 2 est conservé mais un deuxième multiplexeur est interposé
25 entre l'additionneur ADD et le tampon RPC (non représenté). Ce deuxième multiplexeur reçoit en entrée la sortie de l'additionneur ADD et le champ VAL. Sa sortie est appliquée au registre RPC et délivre PC+1, PC+VAL (saut relatif) ou VAL (saut absolu). Deux codes
30 opérations distincts CODE_{JMPC-rel} et CODE_{JMPC-abs} sont prévus pour chaque type de saut conditionnel. Le décodeur PREDEC (fig. 1) ou DEC3 (fig. 2) est agencé pour faire la distinction entre chacun des codes opérations, et commande le multiplexeur MUX et le deuxième multiplexeur
35 en fonction du code reçu.

REVENDECATIONS

1. Microprocesseur pipeline (MP1, MP2) comprenant un compteur ordinal (PC), des moyens (MUX, ADD) d'incrémentation du compteur ordinal (PC), des moyens (PREDEC, DEC1, DEC2, DEC3) de décodage d'un jeu
5 d'instructions du microprocesseur comprenant des moyens (PREDEC, DEC3) de décodage d'une instruction de saut conditionnel (JMPc) du compteur ordinal (PC), une unité de calcul (ALU) comportant une première sortie (S1) pour délivrer un résultat et une deuxième sortie (S2) pour
10 délivrer des bits d'état (C, N, P, Z) du résultat, caractérisé en ce que :

- l'unité de calcul (ALU) et les moyens (PREDEC, DEC3) de décodage de l'instruction de saut conditionnel (JMPc) sont agencés dans deux étages pipeline (ST1, ST2)
15 voisins, et
- les moyens (PREDEC, DEC3) de décodage de l'instruction de saut conditionnel (JMPc) sont connectés à la deuxième sortie (S2) de l'unité de calcul (ALU).

2. Microprocesseur selon la revendication 1,
20 comprenant un premier et un deuxième étages pipeline (ST1, ST2) comprenant chacun un premier et un deuxième secteurs (ST11, ST12, ST21, ST22), le premier secteur (ST11, ST21) de chaque étage étant actif pendant une première demi-période d'horloge (H1) et le deuxième
25 secteur (ST12, ST22) actif pendant une deuxième demi-période d'horloge.

3. Microprocesseur (MP1) selon la revendication 2, dans lequel :

- les moyens (PREDEC) de décodage de l'instruction de saut conditionnel (JMPc) sont agencés dans le premier
30 secteur (ST11) du premier étage pipeline (ST1), et
- l'unité de calcul (ALU) est agencée dans le premier secteur (ST21) du deuxième étage pipeline (ST2).

4. Microprocesseur (MP2) selon la revendication 2, dans lequel :

- les moyens (DEC3) de décodage de l'instruction de saut conditionnel (JMPc) sont agencés dans le deuxième secteur (ST12) du premier étage pipeline (ST1),
- l'unité de calcul (ALU) est agencée dans le premier secteur (ST21) du deuxième étage pipeline (ST2).

5. Microprocesseur selon l'une des revendications 2 à 4, dans lequel le circuit (SPC) d'incrémentation du compteur ordinal (PC) est agencé dans le deuxième secteur (ST12) du premier étage pipeline (ST1).

6. Microprocesseur selon l'une des revendications 2 à 5, comprenant une banque de registres (REGBANK) appartenant en lecture au premier secteur (ST21) et en écriture au deuxième secteur (ST22) du deuxième étage pipeline (ST2).

7. Microprocesseur selon la revendication 6, dans lequel les secteurs (ST21, ST22) du deuxième étage pipeline (ST2) sont délimités par un tampon à trois états (B4) agencé entre les première (S1) et deuxième (S2) sorties de l'unité de calcul (ALU) et des entrées d'écriture (INRd, INRf) de la banque de registre (REGBANK).

8. Microprocesseur (MP1) selon l'une des revendications précédentes, dans lequel :

- l'instruction de saut conditionnel (JMPc) est une instruction de saut relatif,
- les moyens (SPC) d'incrémentation du compteur ordinal (PC) comprennent un circuit additionneur (ADD) recevant en entrée la valeur actuelle du compteur ordinal (PC) et la sortie d'un circuit multiplexeur (MUX),
- le circuit multiplexeur (MUX) est piloté par les moyens (PREDEC, DEC3) de décodage de l'instruction de saut conditionnel (JMPc) et reçoit en entrée une valeur d'incrémentation normale (+1) du compteur ordinal et une

valeur de saut relatif (VAL) du compteur ordinal (PC),
contenue dans l'instruction de saut conditionnel (JMPc).

9. Microprocesseur selon l'une des revendications 1
à 7, dans lequel

- 5 - l'instruction de saut conditionnel (JMPc) est une
instruction de saut absolu,
 - les moyens (SPC) d'incrémentation du compteur ordinal
(PC) comprennent un circuit multiplexeur (MUX) recevant
en entrée la sortie d'un circuit additionneur (ADD) et
- 10 une valeur (VAL) de saut absolu du compteur ordinal (PC)
contenue dans l'instruction de saut conditionnel (JMPc),
 - le circuit multiplexeur (MUX) est piloté par les moyens
de décodage (PREDEC, DEC3) de l'instruction de saut
conditionnel (JMPc), et
- 15 - le circuit additionneur (ADD) reçoit en entrée la
valeur actuelle du compteur ordinal (PC) et une valeur
d'incrémentation normale (+1) du compteur ordinal,

10. Microprocesseur selon l'une des revendications
précédentes, dans lequel les moyens de décodage (PREDEC,
20 DEC3) sont agencés pour décoder une instruction compacte
de saut conditionnel (JMPc) comprenant un code opération
(CODE_{JMPc}), une condition (COND) et une valeur de saut
(VAL).

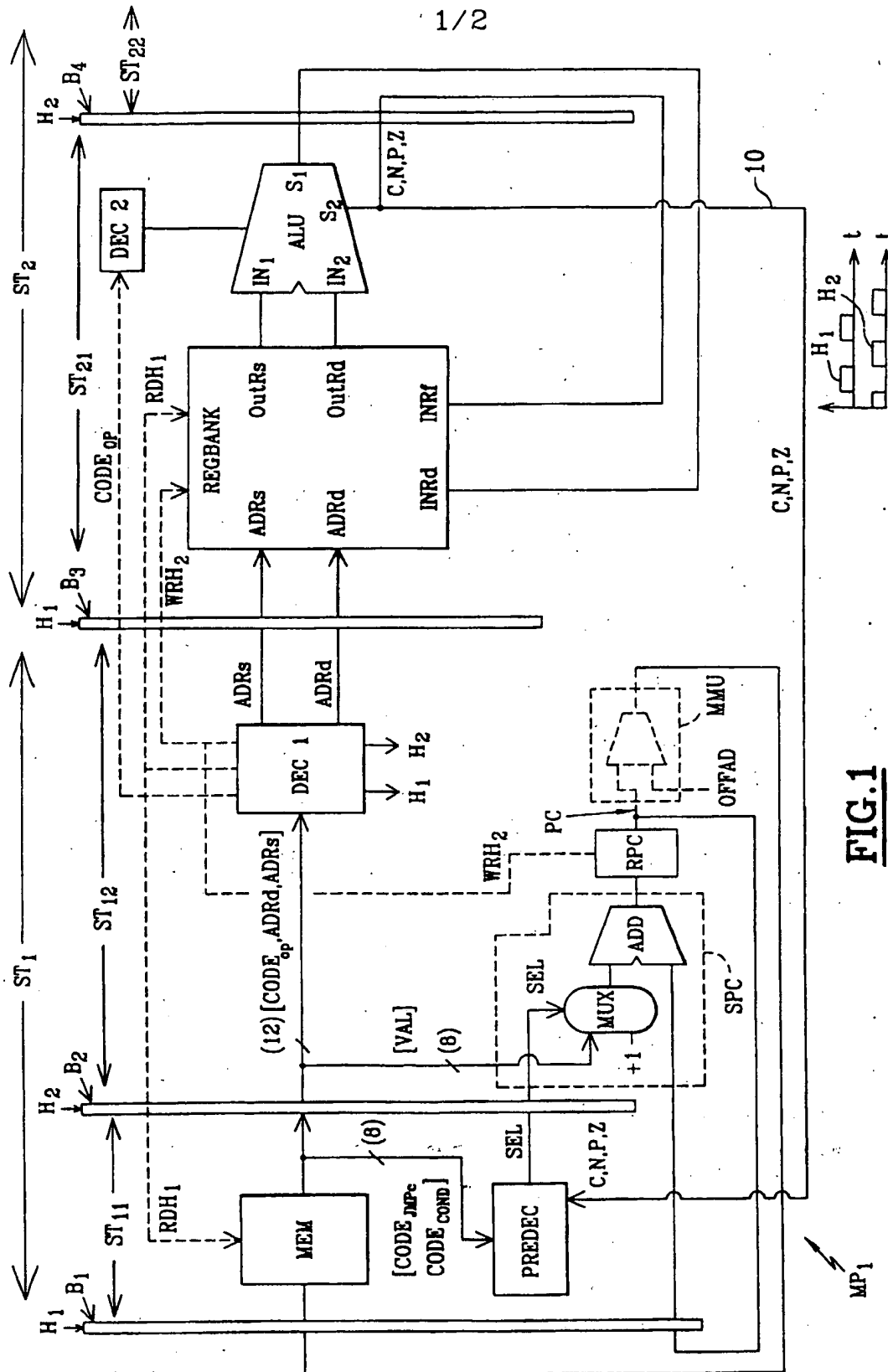


FIG. 1

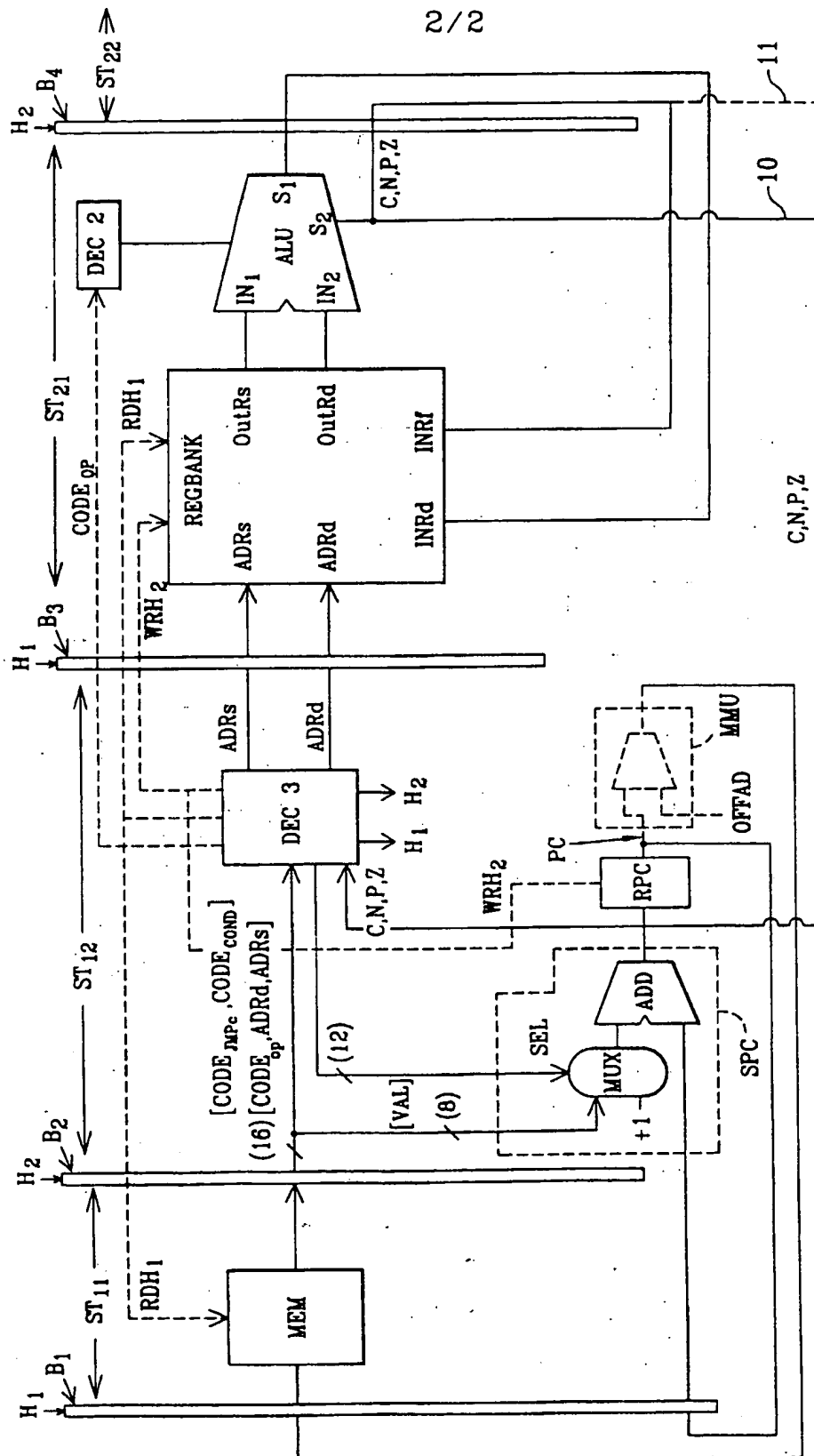


FIG. 2

REPUBLIQUE FRANÇAISE

INSTITUT NATIONAL
de la
PROPRIETE INDUSTRIELLE

RAPPORT DE RECHERCHE
PRELIMINAIRE

établi sur la base des dernières revendications
déposées avant le commencement de la recherche

2770662

N° d'enregistrement
national

FA 552688
FR 9713757

DOCUMENTS CONSIDERES COMME PERTINENTS		Revendications concernées de la demande examinée
Catégorie	Citation du document avec indication, en cas de besoin, des parties pertinentes	
A	PARKER AND VENESKI: "Control structure for making fast conditional branch decisions" IBM TECHNICAL DISCLOSURE BULLETIN, vol. 27, no. 2, juillet 1984, ARMONK,US, pages 1239-1240, XP002070445 * le document en entier *	1
A	US 5 349 671 A (MAEDA TOSHINORI ET AL) 20 septembre 1994 * colonne 7, ligne 56 - colonne 8, ligne 12; figures 4,6 *	1,8
A	MONTANARO J ET AL: "A 160-MHZ, 32-B, 0.5-W CMOS RISC MICROPROCESSOR" IEEE JOURNAL OF SOLID-STATE CIRCUITS, vol. 31, no. 11, novembre 1996, pages 1703-1714, XP000691455 * page 1704, colonne de droite, ligne 33 - page 1705, colonne de droite, ligne 21; figures 3,4 *	1
A	US 5 617 574 A (BOUTAUD FREDERIC ET AL) 1 avril 1997 * colonne 46, ligne 4 - colonne 48, ligne 67; figures 29-35 *	1
A	US 5 485 629 A (DULONG CAROLE) 16 janvier 1996 * colonne 12, ligne 7 - colonne 13, ligne 43; colonne 18, ligne 44 - colonne 19, ligne 55; figures 5,6 *	8,9
A	EP 0 378 816 A (BULL HN INFORMATION SYST) 25 juillet 1990 * colonne 4, lignes 19-28; colonne 6, ligne 16 - colonne 7, ligne 6 *	2-4,6
-/--		
Date d'achèvement de la recherche		Examineur
6 juillet 1998		Klocke, L
<p>CATEGORIE DES DOCUMENTS CITES</p> <p>X: particulièrement pertinent à lui seul Y: particulièrement pertinent en combinaison avec un autre document de la même catégorie A: pertinent à l'encontre d'au moins une revendication ou arrière-plan technologique général O: divulgation non-écrite P: document intermédiaire</p> <p>T: théorie ou principe à la base de l'invention E: document de brevet bénéficiant d'une date antérieure à la date de dépôt et qui n'a été publié qu'à cette date de dépôt ou qu'à une date postérieure. D: cité dans la demande L: cité pour d'autres raisons &: membre de la même famille, document correspondant</p>		

1

EPO FORM 150 03.02 (P4/C13)

REPUBLIQUE FRANÇAISE

INSTITUT NATIONAL
de la
PROPRIETE INDUSTRIELLE

RAPPORT DE RECHERCHE
PRELIMINAIRE
établi sur la base des dernières revendications
déposées avant le commencement de la recherche

2770662

N° d'enregistrement
national

FA 552688
FR 9713757

DOCUMENTS CONSIDERES COMME PERTINENTS		Revendications concernées de la demande examinée
Catégorie	Citation du document avec indication, en cas de besoin, des parties pertinentes	
A	<p>JOUPPI N P: "THE NONUNIFORM DISTRIBUTION OF INSTRUCTION-LEVEL AND MACHINE PARALLELISM AND ITS EFFECT ON PERFORMANCE" IEEE TRANSACTIONS ON COMPUTERS, vol. 38, no. 12, 1 décembre 1989, pages 1645-1658, XP000096380 * page 1647, colonne de gauche, ligne 5 - colonne de droite, ligne 7; figure 2 *</p> <p>-----</p>	2
		DOMAINES TECHNIQUES RECHERCHES (Int.CL.6)
Date d'achèvement de la recherche		Examineur
6 juillet 1998		Klocke, L
<p>CATEGORIE DES DOCUMENTS CITES</p> <p>X : particulièrement pertinent à lui seul Y : particulièrement pertinent en combinaison avec un autre document de la même catégorie A : pertinent à l'encontre d'au moins une revendication ou arrière-plan technologique général O : divulgation non-écrite P : document intermédiaire</p> <p>T : théorie ou principe à la base de l'invention E : document de brevet bénéficiant d'une date antérieure à la date de dépôt et qui n'a été publié qu'à cette date de dépôt ou qu'à une date postérieure. D : cité dans la demande L : cité pour d'autres raisons & : membre de la même famille, document correspondant</p>		

1
EPO FORM 1503 03/92 (P04C13)

THIS PAGE BLANK (USPTO)